```
EEEEEEEEEEEEEEE   RRRRRRRRRRR      FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE   RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE   RRRRRRRRRRRR     FFFFFFFFFFFFFFF
EEE               RRR          RRR  FFF
EEE               RRR          RRR  FFF
EEE               RRR          RRR  FFF
EEE               RRR          RRR  FFF
EEE               RRR          RRR  FFF
EEE               RRR          RRR  FFF
EEEEEEEEEEEE      RRRRRRRRRRRR      FFFFFFFFFFFF
EEEEEEEEEEEE      RRRRRRRRRRRR      FFFFFFFFFFFF
EEEEEEEEEEEE      RRRRRRRRRRRR      FFFFFFFFFFFF
EEE               RRR    RRR        FFF
EEE               RRR    RRR        FFF
EEE               RRR    RRR        FFF
EEE               RRR       RRR     FFF
EEE               RRR       RRR     FFF
EEE               RRR       RRR     FFF
EEEEEEEEEEEEEEE   RRR          RRR   FFF
EEEEEEEEEEEEEEE   RRR          RRR   FFF
EEEEEEEEEEEEEEE   RRR          RRR   FFF
EEEEEEEEEEEEEEE   RRR          RRR   FFF
```

FILEID**LPA11K

```
LL          PPPPPPPP      AAAAAA        11          11      KK      KK
LL          PPPPPPPP      AAAAAA        11          11      KK      KK
LL          PP      PP    AA    AA    1111        1111      KK      KK
LL          PP      PP    AA    AA    1111        1111      KK      KK
LL          PP      PP    AA    AA      11          11      KK    KK
LL          PP      PP    AA    AA      11          11      KK    KK
LL          PPPPPPPP      AA    AA      11          11      KKKKKK
LL          PPPPPPPP      AA    AA      11          11      KKKKKK
LL          PP            AAAAAAAAAA    11          11      KK    KK
LL          PP            AAAAAAAAAA    11          11      KK    KK
LL          PP            AA    AA      11          11      KK      KK      ....
LL          PP            AA    AA      11          11      KK      KK      ....
LLLLLLLLLL  PP            AA    AA    111111      111111    KK      KK      ....
LLLLLLLLLL  PP            AA    AA    111111      111111    KK      KK      ....

LL            IIIIII      SSSSSSSS
LL            IIIIII    SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
0001          SUBROUTINE LPA11K (LUN)
0002    C
0003    C Version:      'V04-000'
0004    C
0005    C*****************************************************************************
0006    C*                                                                           *
0007    C*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0008    C*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0009    C*  ALL RIGHTS RESERVED.                                                     *
0010    C*                                                                           *
0011    C*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0012    C*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0013    C*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0014    C*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0015    C*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0016    C*  TRANSFERRED.                                                             *
0017    C*                                                                           *
0018    C*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0019    C*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
0020    C*  CORPORATION.                                                             *
0021    C*                                                                           *
0022    C*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0023    C*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0024    C*                                                                           *
0025    C*                                                                           *
0026    C*****************************************************************************
0027    C
0028    C
0029    C
0030    C       AUTHOR  BRIAN PORTER            CREATION DATE   20-AUG-1979
0031    C
0032
0033    C++
0034    C       Functional description:
0035    C
0036    C       This module displays entries made for the lpa11k.
0037    C
0038    C       Modified by:
0039    C
0040    C       V03-002 SAR0077         Sharon A. Reynolds,     20-Jun-1983
0041    C               Changed the carriage control in the 'format' statements
0042    C               for use with ERF.
0043    C
0044    C       V03-001 SAR0036         Sharon A. Reynolds,     8-Jun-1983
0045    C               Removed brief/cryptic support.
0046    C
0047    C       v02-004 BP0004          Brian Porter,           23-NOV-1981
0048    C               Minor edit.
0049    C
0050    C       v02-003 BP0003          Brian Porter,           06-NOV-1981
0051    C               Added 'device attention' support.
0052    C
0053    C       v02-002 BP0002          Brian Porter,           23-JUL-1981
0054    C               Added different uba handling.
0055    C
0056    C       v02-001 BP0001          Brian Porter,           29-JUN-1981
0057    C               Removed call to UNUSED_BITS.  Added call to LOGGER and
```

```
0058    C                DHEAD.
0059    C**
0060    C--
0061
0062            INCLUDE 'SRC$:MSGHDR.FOR /NOLIST'
0121            INCLUDE 'SRC$:DEVERR.FOR /NOLIST'
0222
0223
0224            BYTE            LUN
0225            BYTE            MAINTENANCE(0:3)
0226
0227            INTEGER*4       FIELD
0228            INTEGER*4       COMPRESSC
0229            INTEGER*4       CNTRL_MAINT
0230            INTEGER*4       CNTRL_STATUS
0231            INTEGER*4       RDA
0232            INTEGER*4       UBAREG(2)
0233
0234            PARAMETER       TIMEOUT = 96
0235
0236            EQUIVALENCE     (CNTRL_MAINT,EMB$L_DV_REGSAV(0))
0237            EQUIVALENCE     (CNTRL_STATUS,EMB$L_DV_REGSAV(1))
0238            EQUIVALENCE     (RDA,EMB$L_DV_REGSAV(2))
0239            EQUIVALENCE     (MAINTENANCE,EMB$L_DV_REGSAV(3))
0240            EQUIVALENCE     (UBAREG,EMB$L_DV_REGSAV(4))
0241
0242            CHARACTER*7     V1CNTRL_MAINT(0:0)
0243            DATA    V1CNTRL_MAINT(0)/'GO BIT*'/
0244
0245
0246            CHARACTER*20    V2CNTRL_MAINT(6:11)
0247            DATA    V2CNTRL_MAINT(6)/'IN INTERRUPT ENABLE*'/
0248            DATA    V2CNTRL_MAINT(7)/'IN READY*'/
0249            DATA    V2CNTRL_MAINT(8)/'STEP UP*'/
0250            DATA    V2CNTRL_MAINT(9)/'ROM 0*'/
0251            DATA    V2CNTRL_MAINT(10)/'ROM 1*'/
0252            DATA    V2CNTRL_MAINT(11)/'ENABLE ARBITRATION*'/
0253
0254
0255            CHARACTER*11    V3CNTRL_MAINT(13:15)
0256            DATA    V3CNTRL_MAINT(13)/'CRAM WRITE*'/
0257            DATA    V3CNTRL_MAINT(14)/'RESET*'/
0258            DATA    V3CNTRL_MAINT(15)/'RUN*'/
0259
0260
0261            CHARACTER*21    V1CNTRL_STATUS(6:7)
0262            DATA    V1CNTRL_STATUS(6)/'OUT INTERRUPT ENABLE*'/
0263            DATA    V1CNTRL_STATUS(7)/'OUT READY*'/
0264
0265
0266            CHARACTER*31    FATAL_ERROR('340'O:'351'O)
0267            DATA    FATAL_ERROR('340'O)/'SLAVE POWER LOW*'/
0268            DATA    FATAL_ERROR('341'O)/'MASTER FIFO R/W ERROR*'/
0269            DATA    FATAL_ERROR('342'O)/'I/O BUS SACK TIMEOUT*'/
0270            DATA    FATAL_ERROR('343'O)/'MASTER INITIAL CONDITION ERROR*'/
0271            DATA    FATAL_ERROR('344'O)/'MASTER/SLAVE VERSION ERROR*'/
0272            DATA    FATAL_ERROR('345'O)/'SLAVE COLD START TIMEOUT*'/
```

```
0273            DATA    FATAL_ERROR('346'O)/'FATAL SLAVE ERROR*'/
0274            DATA    FATAL_ERROR('347'O)/'FAST PATH DATA ERRORS*'/
0275            DATA    FATAL_ERROR('350'O)/'FIFO DATA ERROR*'/
0276            DATA    FATAL_ERROR('351'O)/'ILLEGAL FATAL ERROR CODE*'/
0277
0278
0279            CHARACTER*33    START_ERROR('300'O:'327'O)
0280            DATA    START_ERROR('300'O)/'NO ROOM FOR REQUEST*'/
0281            DATA    START_ERROR('301'O)/'GO BIT SET WITH ''READY IN'' SET*'/
0282            DATA    START_ERROR('302'O)/'MULTI-USER REQ IN DEDICATED MODE*'/
0283            DATA    START_ERROR('303'O)/'ILLEGAL ''START ERROR'' CODE*'/
0284            DATA    START_ERROR('304'O)/'DEDICATED REQ IN MULTI-USER MODE*'/
0285            DATA    START_ERROR('305'O)/'ILLEGAL ''START ERROR'' CODE*'/
0286            DATA    START_ERROR('306'O)/'START REQUEST WITHOUT INITIALIZE*'/
0287            DATA    START_ERROR('307'O)/'ILLEGAL ''START ERROR'' CODE*'/
0288            DATA    START_ERROR('310'O)/'MULTIPLE INITIALIZE COMMAND*'/
0289            DATA    START_ERROR('311'O)/'ILLEGAL ''START ERROR'' CODE*'/
0290            DATA    START_ERROR('312'O)/'STOP COMMAND AND USER NOT ACTIVE*'/
0291            DATA    START_ERROR('313'O)/'ILLEGAL ''START ERROR'' CODE*'/
0292            DATA    START_ERROR('314'O)/'ODD REQUEST DESCRIPTOR ARRAY ADDR*'/
0293            DATA    START_ERROR('315'O)/'ILLEGAL ''START ERROR'' CODE*'/
0294            DATA    START_ERROR('316'O)/'INITIALIZE WITH IMPROPER VERSION*'/
0295            DATA    START_ERROR('317'O)/'ILLEGAL ''START ERROR'' CODE*'/
0296            DATA    START_ERROR('320'O)/'DEVICE NOT PRESENT*'/
0297            DATA    START_ERROR('321'O)/'ILLEGAL ''START ERROR'' CODE*'/
0298            DATA    START_ERROR('322'O)/'ILLEGAL FUNCTION*'/
0299            DATA    START_ERROR('323'O)/'ILLEGAL FUNCTION*'/
0300            DATA    START_ERROR('324'O)/'REQUEST DESCRIPTOR ARRAY NXM*'/
0301            DATA    START_ERROR('325'O)/'BUFFER/USER STATUS WORD ODD ADDR*'/
0302            DATA    START_ERROR('326'O)/'DEVICE NOT PRESENT ON I/O BUS*'/
0303            DATA    START_ERROR('327'O)/'XFER COUNT LESS THAN 256 WORDS*'/
0304
0305            CHARACTER*33    USER_ERROR('240'O:'261'O)
0306            DATA    USER_ERROR('240'O)/'NON-FATAL ERROR COUNT EXCEEDED*'/
0307            DATA    USER_ERROR('241'O)/'ERROR STATUS OVER-RUN*'/
0308            DATA    USER_ERROR('242'O)/'RANDOM CHANNEL LIST NXM*'/
0309            DATA    USER_ERROR('243'O)/'BUFFER OVER/UNDER -RUN*'/
0310            DATA    USER_ERROR('244'O)/'BUFFER NXM*'/
0311            DATA    USER_ERROR('245'O)/'USER STATUS WORD NXM*'/
0312            DATA    USER_ERROR('246'O)/'USER STATUS WORD INV BUFR INDEX*'/
0313            DATA    USER_ERROR('247'O)/'MASTER FIFO 7/8 FULL*'/
0314            DATA    USER_ERROR('250'O)/'REQ TERM BY USER STAT WORD*'/
0315            DATA    USER_ERROR('251'O)/'NO FIRMWARE FIFO BUFFER*'/
0316            DATA    USER_ERROR('252'O)/'SLAVE FIFO 7/8 FULL*'/
0317            DATA    USER_ERROR('253'O)/'RANDOM CHANNEL ADDRESS UNDER-RUN*'/
0318            DATA    USER_ERROR('254'O)/'DATA UNDER-RUN*'/
0319            DATA    USER_ERROR('255'O)/'CHANNEL/DEVICE NON-EXISTENT*'/
0320            DATA    USER_ERROR('256'O)/'ILLEGAL ''USER ERROR'' CODE*'/
0321            DATA    USER_ERROR('257'O)/'ILLEGAL ''USER ERROR'' CODE*'/
0322            DATA    USER_ERROR('260'O)/'MULT EXT TRIG DIGITAL OUTPUT REQ*'/
0323            DATA    USER_ERROR('261'O)/'ILLEGAL ''USER ERROR'' CODE*'/
0324
0325
0326            CHARACTER*33    STATUS_RETURN(0:3)
0327            DATA    STATUS_RETURN(0)/'SUCCESSFUL INITIALIZED DATA XFER*'/
0328            DATA    STATUS_RETURN(1)/'BUFFER FULL*'/
0329            DATA    STATUS_RETURN(2)/'NON-FATAL BUFFER OVER-RUN*'/
```

```
0330            DATA    STATUS_RETURN(3)/'ILLEGAL ''STATUS RETURN'' CODE*'/
0331
0332
0333
0334            CALL FRCTOF (LUN)
0335
0336            call dhead1 (lun,'UBA LPA11K')
0337
0338            CALL LINCHK (LUN,2)
0339
0340            WRITE(LUN,10) CNTRL_MAINT
0341    10      FORMAT(/' ',T8,'LA CS1',T24,Z8.4)
0342
0343            CALL OUTPUT (LUN,CNTRL_MAINT,V1CNTRL_MAINT,0,0,0,'0')
0344
0345            DO 20,I = 2,3
0346
0347            FIELD = LIB$EXTZV(I,1,CNTRL_MAINT)
0348
0349            IF (FIELD .NE. 0) THEN
0350
0351            CALL LINCHK (LUN,1)
0352
0353            WRITE(LUN,15) I + 14
0354    15      FORMAT(' ',T40,'EXTENDED BUS ADDRESS BIT ',I2,'.')
0355            ENDIF
0356
0357    20      CONTINUE
0358
0359            CALL OUTPUT (LUN,CNTRL_MAINT,V2CNTRL_MAINT,6,6,11,'0')
0360
0361            CALL OUTPUT (LUN,CNTRL_MAINT,V3CNTRL_MAINT,13,13,15,'0')
0362
0363            CALL LINCHK (LUN,1)
0364
0365            WRITE(LUN,25) CNTRL_STATUS
0366    25      FORMAT(' ',T8,'LA CS2',T24,Z8.4)
0367
0368            FIELD = LIB$EXTZV(8,8,CNTRL_STATUS)
0369
0370            IF (FIELD .LT. '300'O) THEN
0371
0372            FIELD = LIB$EXTZV(0,3,CNTRL_STATUS)
0373
0374            CALL LINCHK (LUN,1)
0375
0376            WRITE(LUN,30) FIELD
0377    30      FORMAT(' ',T40,'USER INDEX = ',I1,'.')
0378            ENDIF
0379
0380            CALL OUTPUT (LUN,CNTRL_STATUS,V1CNTRL_STATUS,6,6,7,'0')
0381
0382            FIELD = LIB$EXTZV(8,8,CNTRL_STATUS)
0383
0384            IF (FIELD .NE. 0) THEN
0385
0386            CALL LINCHK (LUN,1)
```

```
0387
0388            IF (FIELD .GE. '340'O) THEN
0389
0390            WRITE(LUN,40) FATAL_ERROR(MIN('351'O,FIELD))
0391    40      FORMAT(' ',T40,A<COMPRESSC (FATAL_ERROR(MIN('351'O,FIELD)))>)
0392
0393            ELSE IF (FIELD .GE. '300'O) THEN
0394
0395            WRITE(LUN,50) START_ERROR(MIN('330'O,FIELD))
0396    50      FORMAT(' ',T40,A<COMPRESSC (START_ERROR(MAX('330'O,FIELD)))>)
0397
0398            ELSE IF (FIELD .GE. '240'O) THEN
0399
0400            WRITE(LUN,60) USER_ERROR(MIN('261'O,FIELD))
0401    60      FORMAT(' ',T40,A<COMPRESSC (USER_ERROR(MIN('261'O,FIELD)))>)
0402
0403            ELSE IF (FIELD .LE. '100'O) THEN
0404
0405            WRITE(LUN,70) STATUS_RETURN(MIN('3'O,FIELD))
0406    70      FORMAT(' ',T40,A<COMPRESSC (STATUS_RETURN(MIN('3'O,FIELD)))>)
0407            ENDIF
0408            ENDIF
0409
0410            CALL LINCHK (LUN,1)
0411
0412            WRITE(LUN,90) RDA
0413    90      FORMAT(' ',T8,'LA RDA',T24,Z8.4)
0414
0415            CALL LINCHK (LUN,1)
0416
0417            WRITE(LUN,100) (MAINTENANCE(I+J),J = 3,0,-1)
0418    100     FORMAT(' ',T8,'LA MR',T24,4Z2.2)
0419
0420            IF (FIELD .EQ. '343'O) THEN
0421
0422            CALL LINCHK (LUN,2)
0423
0424            WRITE(LUN,110) MAINTENANCE(1)
0425    110     FORMAT(' ',T40,'EXPECTED STATUS = 052 (OCTAL)',/,
0426           1 T40,'ACTUAL STATUS   = ',O3,' (OCTAL)')
0427
0428            ELSE IF (FIELD .EQ. '344'O) THEN
0429
0430            CALL LINCHK (LUN,2)
0431
0432            WRITE(LUN,120) (MAINTENANCE(I),I = 0,1)
0433    120     FORMAT(' ',T40,'MASTER VERSION NO. = ',I3,'.',/,
0434           1 T40,'SLAVE VERSION NO.  = ',I3,'.')
0435
0436            ELSE IF (FIELD .EQ. '346'O) THEN
0437
0438            IF (MAINTENANCE(0) .EQ. '200'O) THEN
0439
0440            CALL LINCHK (LUN,1)
0441
0442            WRITE(LUN,125)
0443    125     FORMAT(' ',T40,'SLAVE FIFO R/W ERROR')
```

```
0444
0445            ELSE IF (MAINTENANCE(1) .EQ. '201'O) THEN
0446
0447            CALL LINCHK (LUN,1)
0448
0449            WRITE(LUN,130)
0450     130    FORMAT(' ',T40,'KW11-K CLOCK A OVER-RUN')
0451            ENDIF
0452
0453            CALL LINCHK (LUN,1)
0454
0455            WRITE(LUN,135) MAINTENANCE(1)
0456     135    FORMAT(' ',T40,'SLAVE FIRMWARE SEQ = ',O3,' (OCTAL)')
0457
0458            ELSE IF (FIELD .EQ. '347'O) THEN
0459
0460            CALL LINCHK (LUN,2)
0461
0462            WRITE(LUN,140) (MAINTENANCE(I),I = 0,1)
0463     140    FORMAT(' ',T40,'RETURNED VALUE = ',O3,' (OCTAL)',/
0464            1 T40,'EXPECTED VALUE = ',O3,' (OCTAL)')
0465
0466            ELSE IF (FIELD .EQ. '350'O) THEN
0467
0468            CALL LINCHK (LUN,2)
0469
0470            WRITE(LUN,150) (MAINTENANCE(I),I = 0,1)
0471     150    FORMAT(' ',T40,'RETURNED VALUE = ',O3,' (OCTAL)',/
0472            1 T40,'EXPECTED VALUE = ',O3,' (OCTAL)')
0473
0474            ELSE IF (FIELD .EQ. '322'O) THEN
0475
0476            CALL LINCHK (LUN,1)
0477
0478            WRITE(LUN,155) (MAINTENANCE(I),I = 1,0,-1)
0479     155    FORMAT(' ',T40,'ADDR OF MISSING DEV = ',O6)
0480            ENDIF
0481
0482            if (emb$w_hd_entry .ne. 98) then
0483
0484            if (ubareg(1) .ne. 0) then
0485
0486            call uba_datapath (lun,ubareg(1),ubareg(2))
0487            endif
0488            endif
0489
0490            call linchk (lun,1)
0491
0492            write(lun,160)
0493     160    format(' ',:)
0494
0495            call ucb$l_char (lun,emb$l_dv_char)
0496
0497            call ucb$w_sts (lun,emb$w_dv_sts)
0498
0499            call ucb$l_opcnt (lun,emb$l_dv_opcnt)
0500
```

```
0501            call ucb$w_errcnt (lun,emb$w_dv_errcnt)
0502
0503            if (emb$w_hd_entry .ne. 98) then
0504
0505            call linchk (lun,1)
0506
0507            write(lun,160)
0508
0509            call lpa11k_qio (lun,emb$w_dv_func)
0510
0511            call irp$w_bcnt (lun,emb$w_dv_bcnt)
0512
0513            call irp$w_boff (lun,emb$w_dv_boff)
0514
0515            call irp$l_pid (lun,emb$l_dv_rqpid)
0516
0517            call irp$q_iosb (lun,emb$l_dv_iosb1)
0518            endif
0519
0520            RETURN
0521            END
```

PROGRAM SECTIONS

	Name	Bytes	Attributes
0	$CODE	1756	PIC CON REL LCL SHR EXE RD NOWRT LONG
1	$PDATA	659	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2	$LOCAL	2572	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3	EMB	512	PIC OVR REL GBL SHR NOEXE RD WRT LONG
	Total Space Allocated	5499	

ENTRY POINTS

Address	Type	Name
0-00000000		LPA11K

VARIABLES

Address	Type	Name	Address	Type	Name
3-00000052	I*4	CNTRL_MAINT	3-00000056	I*4	CNTRL_STATUS
3-0000001C	L*1	EMB$B_DV_CLASS	3-00000010	L*1	EMB$B_DV_ERTCNT
3-00000011	L*1	EMB$B_DV_ERTMAX	3-0000003E	L*1	EMB$B_DV_NAMLNG
3-0000003A	L*1	EMB$B_DV_SLAVE	3-0000001D	L*1	EMB$B_DV_TYPE
3-00000036	I*4	EMB$L_DV_CHAR	3-00000012	I*4	EMB$L_DV_IOSB1
3-00000016	I*4	EMB$L_DV_IOSB2	3-00000026	I*4	EMB$L_DV_MEDIA
3-0000004E	I*4	EMB$L_DV_NUMREG	3-0000002E	I*4	EMB$L_DV_OPCNT
3-00000032	I*4	EMB$L_DV_OWNUIC	3-0000001E	I*4	EMB$L_DV_RQPID
3-00000000	I*4	EMB$L_HD_SID	3-0000003F	CHAR	EMB$T_DV_NAME

```
  3-00000024  I*2  EMB$W_DV_BCNT          3-00000022  I*2  EMB$W_DV_BOFF
  3-0000002C  I*2  EMB$W_DV_ERRCNT        3-0000003C  I*2  EMB$W_DV_FUNC
  3-0000001A  I*2  EMB$W_DV_STS           3-0000002A  I*2  EMB$W_DV_UNIT
  3-00000004  I*2  EMB$W_HD_ENTRY         3-0000000E  I*2  EMB$W_HD_ERRSEQ
  2-000007F0  I*4  FIELD                  2-000007F4  I*4  I
  2-000007F8  I*4  J                    AP-00000004@  L*1  LUN
  3-0000005A  I*4  RDA
```

ARRAYS

Address	Type	Name	Bytes	Dimensions
3-00000000	L*1	EMB	512	(0:511)
3-00000052	I*4	EMB$L_DV_REGSAV	420	(0:104)
3-00000006	I*4	EMB$Q_HD_TIME	8	(2)
2-000000CA	CHAR	FATAL_ERROR	310	(224:233)
3-0000005E	L*1	MAINTENANCE	4	(0:3)
2-00000200	CHAR	START_ERROR	792	(192:215)
2-0000076A	CHAR	STATUS_RETURN	132	(0:3)
3-00000062	I*4	UBAREG	8	(2)
2-00000518	CHAR	USER_ERROR	594	(160:177)
2-00000000	CHAR	V1CNTRL_MAINT	7	(0:0)
2-000000A0	CHAR	V1CNTRL_STATUS	42	(6:7)
2-00000007	CHAR	V2CNTRL_MAINT	120	(6:11)
2-0000007F	CHAR	V3CNTRL_MAINT	33	(13:15)

LABELS

Address	Label	Address	Label	Address	Label	Address	Label	Address	Label	Address	Label
1-00000035	10'	1-00000049	15'	**	20	1-0000006F	25'	1-00000082	30'	1-0000009C	40'
1-000000A8	50'	1-000000B4	60'	1-000000C0	70'	1-000000CC	90'	1-000000DF	100'	1-000000F3	110'
1-0000013B	120'	1-0000017C	125'	1-00000198	130'	1-000001B7	135'	1-000001E0	140'	1-00000227	150'
1-0000026E	155'	1-0000028E	160'								

FUNCTIONS AND SUBROUTINES REFERENCED

Type	Name	Type	Name	Type	Name	Type	Name	Type	Name	Type	Name
I*4	COMPRESSC		DHEAD1		FRCTOF		IRP$L_PID		IRP$Q_IOSB		IRP$W_BCNT
	IRP$W_BOFF	I*4	LIB$EXTZV		LINCHK		LPA11R_QIO		OUTPUT		UBA_DATAPATH
	UCB$L_CHAR		UCB$L_OPCNT		UCB$W_ERRCNT		UCB$W_STS				

MAS
061
061
061
061
062
062
062
062
062
062
062
062
062
062
062
063
063
063
063
063
063
063
064
064
064
064
064
064
064
065
065
065
065
065
065
065
065
065
065
066
066
066

```
0001
0002
0003
0004          Subroutine LPA11K_QIO (lun,emb$w_dv_func)
0005
0006
0007          include 'src$:qiocommon.for /nolist'
0271
0272
0273          byte            lun
0274
0275          integer*2       emb$w_dv_func
0276
0277          integer*4       qiocode(0:1,0:63)
0278
0279
0280
0281          if (qiocode(0,0) .eq. 0) then
0282
0283          qiocode(1,01) = %loc(io$_loadmcode)
0284
0285          qiocode(1,02) = %loc(io$_startmproc)
0286
0287          qiocode(1,04) = %loc(io$_initialize)
0288
0289          qiocode(1,05) = %loc(io$_setclockp)
0290
0291          qiocode(1,06) = %loc(io$_startdatap)
0292
0293          qiocode(1,07) = %loc(io$_qstop)
0294
0295          qiocode(1,55) = %loc(io$_setclock)
0296
0297          qiocode(1,56) = %loc(io$_startdata)
0298
0299          do 10,i = 0,63
0300
0301          qiocode(0,i) = 33
0302
0303          if (qiocode(1,i) .eq. 0) then
0304
0305          qiocode(1,i) = %loc(qio_string)
0306          endif
0307
0308    10    continue
0309          endif
0310
0311          call irp$w_func (lun,emb$w_dv_func,
0312          1 qiocode(0,lib$extzv(0,6,emb$w_dv_func)))
0313
0314          return
0315
0316          end
```

PROGRAM SECTIONS

	Name	Bytes	Attributes
0	$CODE	149	PIC CON REL LCL SHR EXE RD NOWRT LONG
1	$PDATA	8	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2	$LOCAL	548	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3	QIOCOMMON	1247	PIC OVR REL GBL SHR NOEXE RD WRT LONG
	Total Space Allocated	1952	

ENTRY POINTS

Address	Type	Name
0-00000000		LPA11K_QIO

VARIABLES

Address	Type	Name	Address	Type	Name
AP-00000008a	I*2	EMB$W_DV_FUNC	2-00000200	I*4	I
3-00000442	CHAR	IO$_ABORT	3-0000034D	CHAR	IO$_ACCESS
3-000003C2	CHAR	IO$_ACPCONTROL	3-000004B3	CHAR	IO$_AVAILABLE
3-00000297	CHAR	IO$_CLEAN	3-00000369	CHAR	IO$_CREATE
3-00000385	CHAR	IO$_DEACCESS	3-00000393	CHAR	IO$_DELETE
3-0000026D	CHAR	IO$_DIAGNOSE	3-00000065	CHAR	IO$_DRVCLR
3-000004C8	CHAR	IO$_DSE	3-000000A9	CHAR	IO$_ERASETAPE
3-00000276	CHAR	IO$_FORMAT	3-00000071	CHAR	IO$_INITIALIZE
3-00000014	CHAR	IO$_LOADMCODE	3-000003A1	CHAR	IO$_MODIFY
3-000003E2	CHAR	IO$_MOUNT	3-00000000	CHAR	IO$_NOP
3-0000009D	CHAR	IO$_OFFSET	3-000000EB	CHAR	IO$_PACKACK
3-000000E0	CHAR	IO$_QSTOP	3-000003EF	CHAR	IO$_RDSTATS
3-00000421	CHAR	IO$_READCSR	3-00000169	CHAR	IO$_READHEAD
3-00000286	CHAR	IO$_READLBLK	3-0000013F	CHAR	IO$_READPBLK
3-00000200	CHAR	IO$_READPRESET	3-00000195	CHAR	IO$_READTRACKD
3-0000033A	CHAR	IO$_READVBLK	3-0000045A	CHAR	IO$_READWTHBUF
3-00000484	CHAR	IO$_READWTHXBUF	3-0000004D	CHAR	IO$_RECAL
3-0000007C	CHAR	IO$_RELEASE	3-000001AB	CHAR	IO$_REREADN
3-000001B8	CHAR	IO$_REREADP	3-000000CA	CHAR	IO$_RETCENTER
3-000002E6	CHAR	IO$_REWIND	3-000002C9	CHAR	IO$_REWINDOFF
3-000000FC	CHAR	IO$_SEARCH	3-00000024	CHAR	IO$_SEEK
3-00000231	CHAR	IO$_SENSECHAR	3-00000309	CHAR	IO$_SENSEMODE
3-0000021D	CHAR	IO$_SETCHAR	3-000003B8	CHAR	IO$_SETCLOCK
3-00000088	CHAR	IO$_SETCLOCKP	3-000002DD	CHAR	IO$_SETMODE
3-000002ED	CHAR	IO$_SKIPFILE	3-000002FA	CHAR	IO$_SKIPRECORD
3-00000029	CHAR	IO$_SPACEFILE	3-0000010E	CHAR	IO$_SPACERECORD
3-000003D7	CHAR	IO$_STARTDATA	3-000000B4	CHAR	IO$_STARTDATAP
3-00000037	CHAR	IO$_STARTMPROC	3-0000020F	CHAR	IO$_STARTSPNDL
3-00000059	CHAR	IO$_STOP	3-0000000D	CHAR	IO$_UNLOAD
3-0000046B	CHAR	IO$_WRITEBUFNCRC	3-0000011E	CHAR	IO$_WRITECHECK
3-000001E4	CHAR	IO$_WRITECHECKH	3-000003FF	CHAR	IO$_WRITECSR
3-00000153	CHAR	IO$_WRITEHEAD	3-000002A2	CHAR	IO$_WRITELBLK
3-00000247	CHAR	IO$_WRITEMARK	3-00000314	CHAR	IO$_WRITEOF
3-0000012A	CHAR	IO$_WRITEPBLK	3-000001C9	CHAR	IO$_WRITERET

```
  3-0000017E  CHAR IO$_WRITETRACKD                3-00000326  CHAR IO$_WRITEVBLK
  3-00000448  CHAR IO$_WRITEWTHBUF                3-00000257  CHAR IO$_WRTTMKR
 AP-00000004@ L*1  LUN                            3-000004A1  CHAR QIO_STRING

ARRAYS

    Address  Type  Name                Bytes  Dimensions

  2-00000000  I*4  QIOCODE               512  (0:1, 0:63)

LABELS

    Address   Label

      **      10

FUNCTIONS AND SUBROUTINES REFERENCED

  Type  Name              Type  Name

        IRP$W_FUNC         I*4  LIB$EXTZV

COMMAND QUALIFIERS

  FORTRAN /LIS=LIS$:LPA11K/OBJ=OBJ$:LPA11K MSRC$:LPA11K
  /CHECK=(NOBOUNDS,OVERFLOW,NOUNDERFLOW)
  /DEBUG=(NOSYMBOLS,TRACEBACK)
  /STANDARD=(NOSYNTAX,NOSOURCE_FORM)
  /SHOW=(NOPREPROCESSOR,NOINCLUDE,MAP)
  /F77  /NOG_FLOATING  /I4  /OPTIMIZE  /WARNINGS  /NOD_LINES  /NOCROSS_REFERENCE  /NOMACHINE_CODE  /CONTINUATIONS=19

COMPILATION STATISTICS

  Run Time:              7.64 seconds
  Elapsed Time:          17.17 seconds
  Page Faults:           280
  Dynamic Memory:        209 pages
```

0150 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

LPA11K
LIS

MCHECKS
LIS

LCPRINTER
LIS

MBA
LIS

LOGMSCP
LIS

LINCHK
LIS

MBAINT
LIS

LABEL
LIS

LOGGER
LIS

MASSTAPE
LIS

MASSDISK
LIS